(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101855653 B
(45)授权公告日 2015.12.02

(21)申请号 200880115904.1
(22)申请日 2008.09.24
(30)优先权数据
07117498.1 2007.09.28 EP
(85)PCT国际申请进入国家阶段日
2010.05.11
(86)PCT国际申请的申请数据
PCT/FI2008/050529 2008.09.24
(87)PCT国际申请的公布数据
WO2009/040470 EN 2009.04.02
(73)专利权人 伊洛克公司
地址 芬兰奥卢
(72)发明人 塞波·洛希尼瓦 米卡·普卡里
(74)专利代理机构 北京集佳知识产权代理有限公司 11227
代理人 康建峰 陈炜
(51)Int.Cl.
G07C 9/00(2006.01)
(56)对比文件
US 2004025039 A1,2004.02.05, 摘要、说明书第[0021]-[0127]段、附图1-7.
审查员 孟祥龙

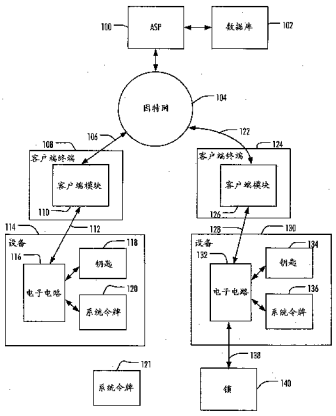
权利要求书2页 说明书9页 附图8页

(54)发明名称

锁管理系统

(57)摘要

提供了一种用于自供电锁的锁管理系统。该系统包括:ASP(应用服务提供商)服务器,在操作上连接到因特网并且被配置成存储锁系统相关信息;至少一个客户端模块,被配置成:控制用于加密和解密的共享机密的生成、以及使用令牌进行的锁访问数据包的生成和加密,使用公共网络将数据包传送给ASP服务器,使用公共网络从ASP服务器接收经加密的状态包,控制状态包的解密,并使用公共网络将关于解密状态包的信息发送给ASP服务器;以及至少一个锁,被配置成:通过公共网络从ASP服务器接收数据包,对数据包解密,并使用公共网络将经加密的状态包发送给ASP服务器。



CN 101855653 B

1. 一种用于自供电锁的锁管理系统，包括：

应用服务提供商服务器、至少一个锁、至少一个客户端模块、第一设备以及系统令牌，其中所述应用服务提供商服务器被配置成存储锁系统相关信息；

所述系统令牌被配置成存储用于对钥匙和锁进行编程的锁定系统共享机密，

所述至少一个客户端模块被配置成：

控制所述第一设备利用系统令牌、通过生成用于加密和解密的共享机密来对钥匙进行编程，

控制所述第一设备利用系统令牌、通过生成用于加密和解密的共享机密来对锁访问包进行编程，

使用公共网络将所述锁访问包传送给所述应用服务提供商服务器，

使用公共网络从所述应用服务提供商服务器接收经加密的状态包，

控制所述状态包的解密，并且

使用公共网络将关于解密状态包的信息发送给所述应用服务提供商服务器；

所述应用服务提供商服务器被配置成：

在操作上连接到因特网，

维护用于存储锁和钥匙访问数据，以及

存储锁访问包和经加密的状态包的数据库，

并且所述至少一个锁被配置成：

通过公共网络从所述应用服务提供商服务器接收数据包，

对所述数据包解密，并使用公共网络将经加密的状态包发送给所述应用服务提供商服务器。

2. 如权利要求 1 所述的锁管理系统，其中，客户端模块被配置成控制所述第一设备生成这样的锁访问数据包：其包括关于锁所属的锁定系统的以及关于所述锁的访问权的信息。

3. 如权利要求 1 所述的锁管理系统，其中，客户端模块被配置成控制所述第一设备生成这样的锁访问数据包：其包括将锁恢复到初始状态的命令。

4. 如权利要求 1 所述的锁管理系统，其中，所述第一设备被配置成与钥匙、客户端模块相连以及与所述系统令牌通信。

5. 如权利要求 1 所述的锁管理系统，包括：第二设备，被配置成与锁相连以及与所述系统令牌通信。

6. 如权利要求 5 所述的锁管理系统，包括：第二客户端模块，被配置成使用公共网络与所述应用服务提供商服务器相连以及通过有线或无线连接与所述第二设备相连。

7. 如权利要求 6 所述的锁管理系统，其中，所述第二客户端模块被配置成从所述应用服务提供商服务器接收锁访问数据包并通过所述第二设备将所述包传送给锁。

8. 如权利要求 6 所述的锁管理系统，其中，所述第二客户端模块被配置成通过所述第二设备从锁接收经加密的状态包并将所述包传送给所述应用服务提供商服务器。

9. 如权利要求 6 所述的锁管理系统，其中，所述第二客户端模块与所述应用服务提供商服务器之间的连接至少部分是无线的。

10. 如权利要求 6 所述的锁管理系统，其中，所述系统包括移动终端中的第二客户端模

块。

11. 一种管理用于自供电锁的系统的方法，包括：

由客户端模块控制第一设备利用系统令牌、通过生成用于加密和解密的共享机密来对钥匙进行编程；

由客户端模块控制第一设备利用系统令牌、通过生成用于加密和解密的共享机密来对锁访问数据包进行编程；

使用所述系统令牌对生成的所述锁访问数据包加密；

使用公共网络将经加密的所述数据包传送给应用服务提供商服务器；

将经加密的所述数据包存储在所述应用服务提供商服务器中；

由锁通过公共网络从所述服务器读取经加密的所述数据包；

在所述锁中对所述数据包解密；

在所述锁中生成经加密的状态包并将所述状态包发送给所述应用服务提供商服务器；

由客户端模块从所述应用服务提供商服务器读取状态包并控制所述状态包的解密；

将关于解密状态包的信息从所述客户端模块传送给所述应用服务提供商服务器。

12. 如权利要求 11 所述的方法，进一步包括：

在客户端模块中生成这样的锁访问数据包：其包括关于锁所属的锁定系统的以及关于所述锁的访问权的信息。

13. 如权利要求 11 所述的方法，进一步包括：

在客户端模块中生成这样的锁访问数据包：其包括锁命令“恢复到初始状态”。

锁管理系统

技术领域

[0001] 本发明涉及用于机电锁的锁管理系统。更具体而言，本发明涉及用于自供电锁的系统。

背景技术

[0002] 各种类型的机电锁正取代传统的机械锁。机电锁需要外部电源、锁内置电池、钥匙内置电池、或者用于在锁内生成电能以使得锁自供电的装置。机电锁与传统锁相比提供了许多优点。它们提供了更好的安全性，并且对钥匙或安全令牌的控制更加容易。

[0003] 另外，大多数机电锁和 / 或钥匙和令牌是可编程的。可以将锁编程为接受不同的钥匙而拒绝其它的钥匙。

[0004] 与机电锁和自供电锁相关联的一个问题是对锁和钥匙的编程。在许多已知的机电锁定系统中，锁制造商将工厂编程的锁交付给最终用户。锁制造商已对属于给定锁定系统的锁执行了所需要的编程。

发明内容

[0005] 根据本发明的一方面，提供了一种用于自供电锁的锁管理系统，包括 :ASP(应用服务提供商) 服务器，在操作上连接到因特网并且被配置成存储锁系统相关信息 ;至少一个客户端模块，被配置成 :控制用于加密和解密的共享机密的生成、以及使用令牌进行的锁访问数据包的生成和加密，使用公共网络将数据包传送给 ASP 服务器，使用公共网络从 ASP 服务器接收经加密的状态包，控制状态包的解密，并使用公共网络将关于解密状态包的信息发送给 ASP 服务器 ;以及至少一个锁，被配置成 :通过公共网络从 ASP 服务器接收数据包，对数据包解密，并使用公共网络将经加密的状态包发送给 ASP 服务器。

[0006] 根据本发明的另一方面，提供了一种管理用于自供电锁的系统的方法，该方法包括 :由客户端模块控制用于加密和解密的共享机密的生成 ;使用安全令牌生成锁访问数据包 ;使用令牌对所生成的锁访问数据包加密 ;使用公共网络将经加密的数据包传送给 ASP(应用服务提供商) 服务器 ;将经加密的数据包存储在 ASP 服务器中 ;由锁通过公共网络从服务器读取经加密的数据包 ;在锁中对数据包解密 ;在锁中生成经加密的状态包并将该包发送给 ASP 服务器 ;由客户端模块从 ASP 服务器读取状态包并控制状态包的解密 ;将关于解密状态包的信息从客户端模块传送给 ASP 服务器。

[0007] 根据本发明的另一方面，提供了一种用于自供电锁的锁管理系统中的客户端模块，该系统包括 ASP(应用服务提供商) 服务器，ASP 服务器在操作上连接到因特网并且被配置成存储锁系统相关信息，客户端模块被配置成 :生成用于加密和解密的共享机密，使用令牌、根据钥匙数据和共享机密生成唯一钥匙机密 ;使用安全令牌生成锁访问数据包并对锁访问数据包加密 ;以及使用公共网络与 ASP 服务器通信。

[0008] 根据本发明的又一方面，提供了一种用于自供电锁的锁管理系统中的锁，该系统包括 ASP(应用服务提供商) 服务器，ASP 服务器在操作上连接到因特网并且被配置成存储

锁系统相关信息；锁被配置成：从 ASP 服务器接收数据包；对数据包解密，使用数据包信息生成共享机密，存储共享机密，并将经加密的状态包发送给 ASP 服务器。

[0009] 本发明具有多个优点。所提出的方案实现了灵活的锁和钥匙编程。锁制造商或经销商维护 ASP 服务器，ASP 服务器维护锁定系统的数据库。然而，锁和钥匙编程由最终用户执行。因此，锁制造商可以交付处于初始状态（其中，锁不属于任何特定锁定系统）的锁。初始状态锁不存储任何安全敏感信息。

[0010] 在所提出的方案中，锁无需具有与 ASP 服务器的专用有线连接。可以将经加密的锁编程数据通过公共网络（可以是有线或无线连接）传送给锁。

附图说明

[0011] 下面，参照附图、仅通过举例来描述本发明的多个实施例，在附图中：

[0012] 图 1 示出了锁管理系统的结构的一个例子；

[0013] 图 2 示出了钥匙和锁；

[0014] 图 3A 是示出了用于生成锁定系统共享机密的实施例的流程图；

[0015] 图 3B 是示出了用于将额外系统令牌创建到锁定系统中的实施例的流程图；

[0016] 图 3C 是示出了用于将锁定系统共享机密传递到锁中的实施例的流程图；

[0017] 图 3D 是示出了用于将钥匙共享机密设置到新钥匙的实施例的流程图；

[0018] 图 3E 是示出了将要使用钥匙打开锁的实施例的流程图；

[0019] 图 4 是示出了本发明一个实施例的信令图；以及

[0020] 图 5 示出了钥匙和锁的另一个例子。

具体实施方式

[0021] 下面的实施例是示例性的。虽然说明书可能会在多个地方引用“一个 (an)”、“一个 (one)”或“一些”实施例，但这不一定意味着每个这样的引用是对相同实施例进行的或者特征只适用于单个实施例。亦可以将不同实施例的特征相组合以提供其它实施例。

[0022] 参照图 1 说明锁管理系统的结构的一个例子。该系统包括在操作上连接到因特网 104 并且被配置成将锁系统相关信息存储到数据库 102 的应用服务提供商 (ASP) 服务器 100。数据库 102 可以利用可拆卸或固定大容量存储器在该服务器中实现，或者可以是单独的计算机。其它的实现也是可行的。通常，锁系统制造商或锁系统经销商维护 ASP 服务器 100。数据库维护与属于锁定系统的锁和钥匙有关的数据。数据包括例如与锁和钥匙身份、钥匙保持器、锁和钥匙状态以及访问权有关的信息。

[0023] 该系统进一步包括客户端模块 110。客户端模块 110 可以是在客户端场所的客户端终端 108 中运行的客户端软件。通常，客户端终端 108 是通过有线或无线连接 106 连接到因特网 104 的个人计算机或相应处理单元。

[0024] 客户端模块 110 的实施可以取决于客户端终端设计而变化。客户端模块可以包括用编程语言（可以是高级编程语言，如 C、Java 等，或者是低级编程语言，如机器语言或汇编语言）编码的程序指令。

[0025] 客户端模块 110 可以被配置成管理锁定系统相关信息。例如，客户端模块可以生成用于加密和解密的共享机密，并且可以使用安全令牌生成锁访问数据包并对锁访问数据

包加密。

[0026] 客户端模块可以连接 112 到第一设备 114，第一设备 114 被配置成与钥匙 118 和系统令牌 120 相连。客户端模块与第一设备之间的连接 112 可以利用有线或无线连接来实现。该连接可以利用 USB、蓝牙、红外线或其它已知无线技术来实现。

[0027] 第一设备 114 包括电子电路 116 以及用于钥匙 118 和令牌 120 的保持器。电子电路 116 可以包括处理器以及用于存储用于处理器的数据和软件的存储器。电子电路可以被配置成执行涉及锁定数据的计算以及在客户端模块、钥匙与系统令牌之间传递信息。第一设备 114 和客户端终端 108 提供了用于客户端模块 110 和钥匙 118 和系统令牌 120 通信的平台。客户端模块 110 和 ASP 服务器 100 与系统令牌 120 通信，以便存储锁系统的共享机密、对锁访问数据包加密和解密以及认证锁系统中的用户访问。

[0028] 锁管理系统可以进一步包括第二客户端模块 126。第二客户端模块 126 可以是在客户端终端 124 中运行的客户端软件。客户端终端 124 可以是连接 122 到因特网 104 的个人计算机、个人数据助理 (PDA) 或移动电话。第二客户端模块 126 可以与客户端模块 110 以相同的方式实施。

[0029] 第二客户端模块 126 可以连接 128 到第二设备 130，第二设备 130 被配置成与钥匙 134 和系统令牌 136 相连。第二客户端模块与第二设备之间的连接 128 可以利用有线或无线连接来实现。该连接可以利用 USB、蓝牙、红外线或其它已知无线技术来实现。另外，第二设备可以具有与锁 140 的连接 138。该连接可以是有线的或无线的。例如，有线连接可以利用 1 线式总线连接来实现。有线连接可以向自供电锁提供电能。无线连接可以利用已知无线协议来实现。

[0030] 第二设备 130 和客户端终端 124 提供了用于客户端模块 126、钥匙 134、系统令牌 136 和锁 140 通信的平台，以便存储锁系统的共享机密、对锁访问数据包加密和解密以及认证锁系统中的用户访问。

[0031] 在一个实施例中，第一设备和第二设备是相同的设备。

[0032] 在一个实施例中，客户端模块 110 或 126 的用户通过登录到 ASP 服务器 100 来建立客户端模块与 ASP 服务器 100 之间的会话。客户端模块可以联系 ASP 服务器并检查是否存在可用模块的更新版本。如果存在，则可以下载该更新版本并将其安装在客户端终端上。在启动或执行了所需要的锁定系统管理操作之后，可以通过登出 ASP 服务器结束会话。

[0033] 图 2 示出了钥匙 118 和锁 140。锁 140 被配置成从钥匙 118 读取访问数据并将数据与预定准则匹配。钥匙 118 包括被配置成存储访问数据以及执行涉及加密和解密的计算的电子电路。电子电路可以例如是 MaximIntegrated Products 的**iButton®** (www.ibutton.com)；这种电子电路可以用**1-Wire®**协议读取。电子电路可以放置在例如钥匙或令牌中，但是也可以置于其它合适的设备或物体中。只要锁可以从电子电路读取数据即可。从钥匙到锁 140 的数据传递可以利用任何合适的有线或无线通信技术来执行。在自供电锁中，所产生的能量的量可能会限制所使用的技术。磁条技术或智能卡技术也可以用在钥匙中。无线技术可以包括例如 RFID（射频识别）技术或移动电话技术。钥匙可以包括应答器、RF 标签或者能够存储数据的任何其它合适的存储器类型。

[0034] 通过将从钥匙读取的数据与预定准则匹配，来使用该数据进行认证。认证可以利用由国家安全局 (NSA) 设计的 SHA-1（安全散列 (hash) 算法）函数来执行。在 SHA-1 中，

根据给定输入数据序列（称为消息）计算出精简数字表示（称为消息摘要）。消息摘要很可能对于消息是唯一的。SHA-1 被称为“安全的”是因为：对于给定算法，寻找对应于给定消息摘要的消息或者寻找产生相同消息摘要的两个不同消息在计算上是不可行的。对消息的任何改变都很可能会导致不同的消息摘要。如果需要提高安全性，则可以使用 SHA 族中的其它散列函数 (SHA-224、SHA-256、SHA-384 和 SHA-512)，它们每个都具有较长的摘要，统称为 SHA-2。当然，可以使用任何合适的认证技术来认证从外部源读取的数据。认证技术的选择取决于锁 140 的期望安全等级，并且可能还取决于尤其是用户供电的机电锁中的用于认证的允许电力消耗。

[0035] 图 3A 是示出了用于生成锁定系统共享机密 (SS) 并将第一系统令牌创建到锁定系统中的实施例的流程图。在对锁访问数据加密和解密时使用锁定系统共享机密。系统令牌包括上述电子电路，并且它被用在第一设备 114 中以生成并存储锁定系统共享机密。系统令牌是特殊令牌，因为它不用作钥匙而是用于对锁定系统的钥匙和锁进行编程。通常，创建系统令牌是对新锁定系统的锁和钥匙进行编程的第一个步骤。锁定系统可以具有一个以上的系统令牌，但它们都存储相同的锁定系统共享机密。

[0036] 客户端模块 110 负责控制系统令牌和锁定系统共享机密的生成。由于客户端模块处于客户端终端中，所以可以在客户端的场所执行该流程，只要客户端模块能访问因特网并且设备 114 连接到客户端终端 108。在一个实施例中，客户端模块 110 控制设备 114 执行下面被分配给客户端模块的任务中的一些或全部。锁制造商或经销商除了维护 ASP 服务器 100 以外不参与该过程。

[0037] 该过程开始于步骤 300，这时用户将空令牌 120 设置到第一设备 114 中。

[0038] 在步骤 302 中，客户端模块 110 请求用户输入种子 1。种子 1 通常可以是具有 10-20 个字符的字母数字串。种子 1 不存储在系统中。用户必须记住它。

[0039] 在步骤 304 中，客户端模块 110 使用随机数发生器生成种子 2。种子 2 通常是 10 至 20 字节长度的数字列表。每个字节可以具有 0 至 255 之间的任何值。

[0040] 在步骤 306 中，客户端模块 110 使用随机发生器生成种子 3。种子 3 通常是 10 至 20 字节长度的。每个字节可以具有 0 至 255 之间的任何值。

[0041] 在步骤 308 中，客户端模块 110 将种子 1-3 发送给令牌 120。令牌 120 接收种子并生成待用作锁定系统共享机密的 SHA-1 散列。令牌 120 将共享机密存储到它的隐藏只写存储器中。不将共享机密回传给客户端模块或展示给用户。

[0042] 如本领域技术人员所熟知，散列可以利用某种其它密码散列函数来生成。本文中仅用 SHA-1 作为例子。

[0043] 在一个实施例中，客户端模块 110 被配置成计算被用作共享机密的散列并将散列发送给存储散列的令牌 120。

[0044] 在步骤 310 中，客户端模块 110 将种子 3 存储在令牌 120 中。

[0045] 在步骤 312 中，客户端模块 110 将种子 2 传送给由 ASP 服务器维护的锁定系统数据库 102。此传送可以例如用 SSL（安全套接层）加密。

[0046] 在步骤 314 中，客户端模块 110 将令牌 120 作为系统令牌注册在锁定系统数据库 102 中。每个令牌可以具有唯一的序列号，其可以存储在数据库 102 中。此存储可以例如用 SSL（安全套接层）加密。

[0047] 该过程结束于 316。

[0048] 图 3B 是示出了用于将额外系统令牌创建到锁定系统中的实施例的流程图。锁定系统已经具有利用图 3A 中所述的流程创建的至少一个系统令牌。客户端模块 110 负责控制额外系统令牌的生成。由于客户端模块处于客户端终端中，所以可以在客户端的场所执行该流程，只要客户端模块能访问因特网并且设备 114 连接到客户端终端 108。在一个实施例中，客户端模块 110 控制设备 114 执行下面被分配给客户端模块的任务中的一些或全部。锁制造商或经销商除了维护 ASP 服务器 100 以外不参与该过程。

[0049] 该过程开始于步骤 320，这时用户使已有系统令牌 120 之一安装在设备 114 中。

[0050] 在步骤 322 中，客户端模块 110 请求用户输入种子 1。种子 1 必须与生成第一系统令牌 120 时输入的种子 1 完全相同。

[0051] 在步骤 324 中，客户端模块 110 通过因特网联系锁系统数据库 102，并从数据库 102 读取种子 2。

[0052] 在步骤 326 中，客户端模块 110 从安装在设备 114 中的已有系统令牌 120 读取种子 3。

[0053] 在步骤 328 中，客户端模块 110 使用种子 1 至 3 并生成 SHA-1 散列。

[0054] 在步骤 330 中，客户端模块 110 使用已有系统令牌 120 验证散列。

[0055] 在步骤 332 中，分析验证结果。如果验证失败，则用户有可能输入了不正确的种子 1，该过程被取消或者从步骤 322 重新开始。

[0056] 否则，该过程接下来进入步骤 334，其中，客户端模块请求用户将已有系统令牌 120 从设备 114 中移除并将空令牌 121 设置到设备 114 中。

[0057] 在步骤 336 中，客户端模块 110 将种子 3 存储在新令牌 121 中。

[0058] 在步骤 338 中，客户端模块 110 将种子 1 和 2 发送给令牌 120。令牌 120 接收种子并使用种子 1 至 3 生成 SHA-1 散列。所生成的散列是锁定系统共享机密，与存储在第一系统令牌 120 中的共享机密相同。令牌将散列作为共享机密存储在它的隐藏只写存储器中。

[0059] 在步骤 340 中，客户端模块 110 将新的系统令牌 121 注册到锁定系统数据库 102 中。此传送可以例如用 SSL（安全套接层）加密。

[0060] 该过程结束于 342。

[0061] 图 3C 是示出了用于将锁定系统共享机密传递到锁中的实施例的流程图。

[0062] 该过程开始于步骤 350，这时用户使已有系统令牌 120 之一安装在设备 114 中。客户端模块 110 再次负责初始步骤。由于客户端模块 110 处于客户端终端 108 中，所以可以在客户端的场所执行该流程，只要客户端模块 110 能访问因特网并且设备 114 连接到客户端终端 108。初始步骤 350 至 366 可以在不同于锁所在地点的地点执行。锁制造商或经销商除了维护 ASP 服务器 100 以外不参与该过程。在一个实施例中，客户端模块 110 控制设备 114 执行下面被分配给客户端模块的任务中的一些或全部。

[0063] 在步骤 352 中，客户端模块 110 请求用户输入种子 1。种子 1 必须与生成第一系统令牌 120 时输入的种子 1 完全相同。

[0064] 在步骤 354 中，客户端模块 110 通过因特网联系锁系统数据库 102，并从数据库 102 读取种子 2。

[0065] 在步骤 356 中，客户端模块 110 从安装在设备 114 中的系统令牌 120 读取种子 3。

[0066] 在步骤 358 中，客户端模块 110 使用种子 1 至 3 并生成 SHA-1 散列。该散列对应于锁定系统的共享机密。

[0067] 在步骤 360 中，客户端模块 110 对照安装在设备 114 中的系统令牌 120 中存储的共享机密来验证散列。

[0068] 在步骤 362 中，分析验证结果。如果验证失败，则用户有可能输入了不正确的种子 1，该过程被取消或者从步骤 332 重新开始。

[0069] 否则，该过程接下来进入步骤364，其中，种子1至3被加密并存储在系统令牌中作为对锁的编程作业。

[0070] 在步骤 366 中，将系统令牌 120 从连接到客户端模块 110 的设备 114 中移除。

[0071] 该流程的其余步骤在锁的安装地点执行。客户端终端 124 包括第二客户端模块 126。客户端终端可以是个人计算机、PDA、智能电话或相应装备。第二设备 130 连接到客户端终端并连接到第二客户端模块，并且它具有与锁 140 的连接。

[0072] 在步骤 368 中，将系统令牌 120（在图 1 中示出为令牌 132）插入与锁 140 相连的设备 130 中。

[0073] 在步骤 370 中，锁 140 从系统令牌 120 读取编程作业，对种子 1 至 3 解密并生成 SHA-1 散列。

[0074] 在步骤 372 中，锁 140 对照安装在设备 130 中的系统令牌 120 中存储的共享机密来验证散列。

[0075] 在步骤 374 中，分析验证结果。

[0076] 如果验证失败，则在步骤 378 中，锁 140 设置“错误”而不设置锁定系统共享机密。

[0077] 如果验证成功，则在步骤 378 中，将共享机密存储在锁 140 中。

[0078] 该过程结束于 376 或 378。

[0079] 可以对多个锁重复步骤 368 至 378。可以利用相同的初始步骤将锁定系统共享机密传递给多个锁。

[0080] 图 3D 是示出了用于将钥匙共享机密设置到新钥匙的实施例的流程图。客户端模块 110 负责控制共享机密的生成。由于客户端模块处于客户端终端中，所以可以在客户端的场所执行该流程，只要客户端模块能访问因特网并且设备 114 连接到客户端终端 108。锁制造商或经销商除了维护 ASP 服务器 100 以外不参与该过程。在一个实施例中，客户端模块 110 控制设备 114 执行下面被分配给客户端模块的任务中的一些或全部。

[0081] 该过程开始于步骤 380，这时新钥匙 118 和已有系统令牌 120 连接在设备 114 中。

[0082] 在步骤 382 中，客户端模块 110 从钥匙 118 读取钥匙数据并将钥匙数据发送给系统令牌 120。钥匙数据可以包括钥匙序列号。

[0083] 在步骤 384 中，系统令牌 120 使用钥匙数据和锁定系统共享机密来计算钥匙共享机密。

[0084] 在步骤 386 中，客户端模块 110 将钥匙共享机密设置到新钥匙 118。

[0085] 在步骤 387 中，客户端模块 110 将新钥匙 188 注册到锁系统数据库 102 中。此传送可以例如用 SSL（安全套接层）加密。

[0086] 该过程结束于 388。

[0087] 除此以外，还可以将额外的访问数据编程到锁定系统的钥匙中。在一个实施例中，

钥匙存储包括钥匙标识、钥匙共享机密和访问组数据的数据结构。每个钥匙具有可以用来标识钥匙的唯一标识 ID。访问组数据包括钥匙所属的一个或多个访问组。

[0088] 在一个实施例中，钥匙可以在它属于被允许进行访问的访问组的情况下或者在钥匙具有被允许进行访问的钥匙标识 ID 的情况下打开锁。

[0089] 利用访问组，大大增强了钥匙的组织。一个钥匙可以设有多个访问组以允许对不同地点的访问。例如，同一个钥匙可以提供对公寓（访问组 1）、地下室（访问组 2）、车库（访问组 3）和垃圾室（访问组 4）的访问。用户于是可以向垃圾管理公司提供只包括访问组 4 的钥匙。由此，可以向该公司提供对垃圾室的访问但该钥匙不授权对建筑物其它部分的访问。

[0090] 图 3E 是示出了将要使用钥匙 118 打开锁 140 的实施例的流程图。

[0091] 该过程开始于步骤 390，这时用户将钥匙 118 插入锁 140 中。在此阶段，自供电锁可以在钥匙被插入锁中时根据钥匙移动生成电能。可替选地，锁可以包括电池。

[0092] 在步骤 391 中，锁 140 从钥匙 118 读取钥匙数据和散列。

[0093] 在步骤 392 中，锁 140 使用存储在锁中的锁定系统共享机密以及钥匙数据来计算 SHA-1 散列。

[0094] 在步骤 393 中，锁 140 对照从钥匙 118 读取的散列来验证由锁计算出的散列。

[0095] 在步骤 394 中，分析验证结果。

[0096] 在步骤 399 中，如果验证失败，则锁 140 设置“错误”而不打开，该过程结束。

[0097] 如果验证成功，则在步骤 396 中，锁 140 验证钥匙访问数据。

[0098] 在步骤 397 中，分析验证结果。钥匙访问数据泄漏钥匙所属的可能访问组的信息。锁检查钥匙所属的访问组与锁被编程为打开的访问组之间是否存在匹配。

[0099] 如果验证失败，则锁 140 设置“错误”而不打开。这在步骤 399 中完成。

[0100] 如果验证成功，则在步骤 398 中，打开锁 140。

[0101] 该过程结束于步骤 398 或 399。

[0102] 图 4 示出了由用户使用客户端模块 110 改变锁 140 的访问权的一个例子。客户端模块 110 负责控制访问权改变的初始部分。由于客户端模块处于客户端终端 108 中，所以可以在客户端的场所执行该流程，只要客户端模块能访问因特网。在该过程开始之前，将系统令牌 120 放置在设备 114 中，并且使设备 114 连接到客户端终端 108 和客户端模块 110。另外，客户端模块登录到 ASP 服务器 100。

[0103] ASP 服务器 100 维护数据库 102（其中存储了与锁定系统的锁、钥匙和访问权有关的信息）。然而，不能在 ASP 服务器改变访问权。访问权的改变需要使用客户端模块 110、126 以及通过设备 114、130 连接到客户端模块的系统令牌。

[0104] 在一个实施例中，客户端模块向系统的用户提供用以改变访问权以及对锁和钥匙进行编程的接口。客户端模块 110 被配置成从用户接收新的锁访问数据。在接收到该数据时，客户端模块 110 将“对锁编程”消息 402 发送给由 ASP 服务器 100 维护的数据库 102。

[0105] ASP 服务器 100 将接收到的数据存储到数据库 102 中，并将经修改的锁访问数据作为“发送作业”消息 404 发回给客户端模块 110。客户端模块 110 接收该消息，并将该数据作为“对作业加密”消息 406 发送给连接到设备 114 的系统令牌 120。系统令牌 120 用锁定系统共享机密对访问数据加密，并将经加密的锁访问数据作为“发送经加密的作业”消息

408 发送给客户端模块 110。客户端模块接收经加密的数据并将它作为“发送经加密的作业”消息 410 发送给 ASP 服务器 100。ASP 服务器 100 将该数据放置到工作队列 400 中，工作队列 400 是数据库 102 的一部分。工作队列 400 是稍后要被传送给锁的经加密的访问数据消息的列表。客户端模块 110 可以登出 ASP 服务器 100。

[0106] 该流程的其余步骤在锁的安装地点执行。首先，用户从客户端模块 126 登入 ASP 服务器 100。应用户的命令，客户端模块联系 ASP 服务器，并利用消息 412 从工作队列 400 中选择对锁编程的作业。工作队列 400 通过将经加密的锁访问数据置于消息 414 中予以发送来进行回复。客户端模块 126 接收作业并将它存储在客户端终端 124 的存储器中。对作业数据所包含的锁访问数据加密，并且，将该数据存储在客户端终端 124 中无安全风险。

[0107] 接下来，将系统令牌 136 放置到设备 130 中。建立设备 130 和客户端终端 124 和客户端模块 126 之间的连接。客户端模块被配置成当从用户接收到“对锁编程”命令时将经加密的锁访问数据 416 发送给系统令牌 136。用户将设备 130 连接到要被编程的锁 140。当锁 140 检测到与设备 130 的连接已建立时，锁被配置成向系统令牌 136 请求 418 锁访问数据。在一个实施例中，锁被配置成在请求数据之前认证系统令牌。

[0108] 系统令牌 136 通过发送经加密的数据 420 进行回复。锁 140 对数据解密，并使用存储在锁中的共享机密来验证其签名。如果数据有效，则锁 140 存储数据并将包括锁编程状态的经加密的确认消息 422 发送给系统令牌 136 以表明锁的访问数据已被编程。如果数据无效，则锁 140 忽略数据并将否认 422 发送给系统令牌 136 以表明锁编程失败。在一个实施例中，设备 130 被配置成利用视觉指示（比如绿色或红色 LED）通知用户锁编程成功。

[0109] 系统令牌 136 将经加密的锁编程状态 424 发送给客户端模块 126。客户端模块 126 将经加密的锁编程状态 426 发送给工作队列 400。

[0110] 锁编程状态保持在工作队列 400 中，直到连接到系统令牌 120 的客户端模块建立与 ASP 服务器 100 的会话为止。客户端模块可以被配置成当连接到 ASP 服务器 100 时检查 428 工作队列 400。作为对查询消息 428 的响应，ASP 服务器 100 将经加密的锁编程状态发送 430 给客户端模块 110。

[0111] 当接收到经加密的状态消息 430 时，客户端模块 110 将该消息发送 432 给系统令牌 120，系统令牌 120 对该数据解密并通过将经解密的数据 434 发送给客户端模块 110 来进行回复。客户端模块将包括锁 140 状态的数据 436 发送给 ASP 服务器 100，ASP 服务器 100 将锁状态存储在数据库 102 中。

[0112] 结合图 3C 描述的流程将锁定系统共享机密安装到锁中。在锁定系统共享机密被安装之前，锁可以处于初始状态。初始状态的锁尚不属于任何锁定系统。它不被配置成认证任何钥匙以及验证钥匙的访问数据。还可以按与图 3C 的流程类似的流程将锁定系统共享机密从锁中移除。在一个实施例中，客户端模块 110 被配置成生成包括将锁恢复到初始状态的命令的锁访问数据包。在共享机密被卸载之后，锁再次回到初始状态并且它可被重新用在其它锁定系统中而无任何安全风险。没有锁定系统共享机密的锁不具有任何存储的安全敏感信息。

[0113] 当使用图 3C 的流程将锁定系统共享机密安装到锁中时，锁是锁定系统的成员。只有属于锁定系统的钥匙才能打开锁。然而，锁不验证任何额外的访问数据。锁的这种状态可以称为交付使用状态。

[0114] 利用如图 3A 中所述的客户端模块 110 或设备 114 中的系统令牌 120、基于用户给予的种子来生成锁定系统共享机密。将锁定系统共享机密存储在系统令牌中只写存储器中。

[0115] 属于由所述锁管理系统管理的系统的锁具有计算锁定系统共享机密作为系统令牌的能力。钥匙具有根据锁定系统共享机密以及每个钥匙的唯一标识生成的唯一机密。锁被配置成基于存储在锁中的锁定系统共享机密以及从钥匙读取的唯一标识来生成钥匙机密。

[0116] 当使用图 4 中所述的流程将锁访问组安装到锁中时，锁能够认证钥匙并验证钥匙访问数据。可以将锁的这种状态描述为工作状态。欧洲专利申请 07112675（其合并于此作为参考）中进一步解释了钥匙访问数据验证。

[0117] 图 5 示出了钥匙 118 和锁 140 的一个例子。在图 5 的例子中，钥匙 118 包括与接触装置 502 和钥匙框架相连的电子电路 500。电子电路 500 可以包括存储单元。图 1 的机电锁 140 是自供电锁。锁 140 包括输能机构 504，输能机构 504 在钥匙 118 被插入锁 140 中时将来自用户的机械能转换给为电子电路 508 供电的发电机 506。在本例子中，电子电路 508 被配置成通过钥匙的接触装置 502 以及接触装置 510 与钥匙的电子电路 500 通信。该通信可以实现为无线连接或者通过物理传导来实现。

[0118] 电子电路 508 被配置成在钥匙插入时从钥匙 118 的电子电路 500 读取钥匙数据。电子电路 508 被进一步配置成如上所述那样认证钥匙并验证访问数据。电子电路可以包括处理器以及用于存储用于处理器的数据和所需软件的存储单元。软件可以被配置成执行与生成锁定系统共享机密、更新访问数据和认证钥匙相关的前述流程。

[0119] 图 5 的锁进一步包括被配置成接收打开命令以及将锁设置于机械可打开状态的致动器 512。致动器 512 可以由利用发电机 506 产生的电能供电。致动器 512 可以被机械地设置于锁定状态，但是阐明本实施例无需对其进行详细讨论。

[0120] 当致动器 512 已将锁设置于机械可打开状态时，可以通过例如旋转钥匙 118 来移动锁簧机构 514。所需机械能也可以由用户通过转动门的旋钮或把手（图 5 中未示出）来产生。也可以使用其它合适的转动机构。

[0121] 上述步骤和相关功能并非绝对按时间顺序，其中一些步骤可以同时执行或者按不同于给定顺序的顺序执行。也可以在步骤之间或在步骤内执行其它功能。其中一些或一部分步骤也可以被省去，或者由相应步骤或该步骤的一部分代替。

[0122] 对于本领域技术人员而言明显的是，随着技术的进步，可以按各种方式实施本发明的构思。本发明及其实施例不限于上述例子，而是可以在权利要求的范围内变化。

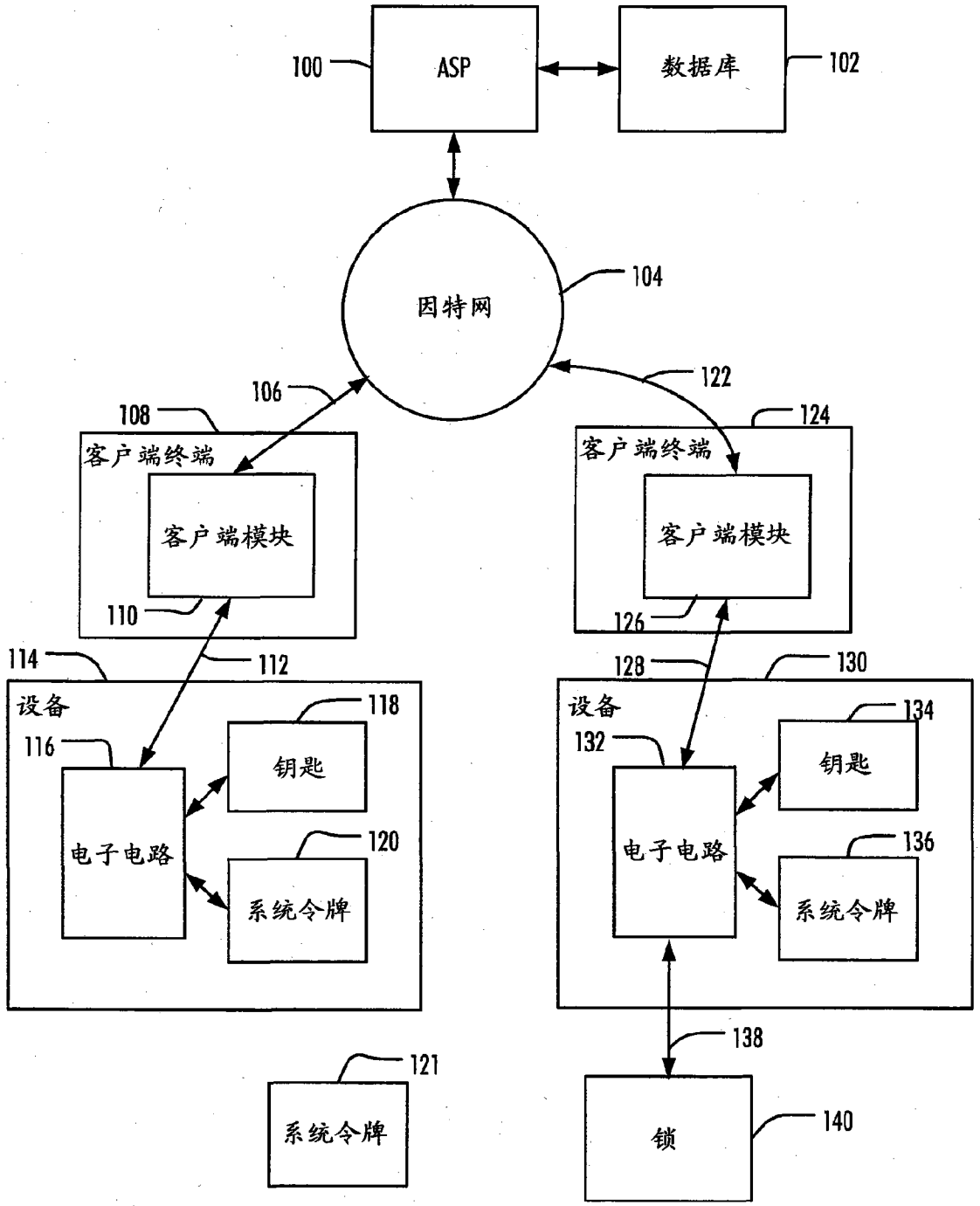

100
ASP
数据库
102
因特网
104
106
122
108
客户端终端
客户端模块
110
124
客户端终端
客户端模块
126
114
112
128
130
设备
116
118
钥匙
电子电路
120
系统令牌
设备
132
134
钥匙
电子电路
136
系统令牌
121
系统令牌
138
锁
140

图 1

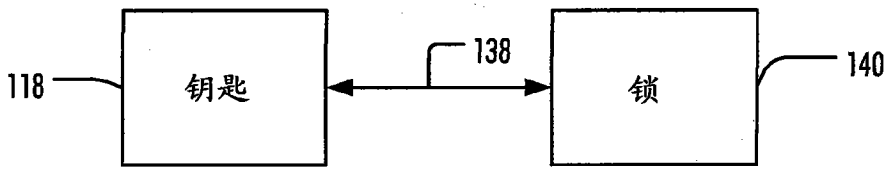

118
钥匙
138
锁
140

图 2

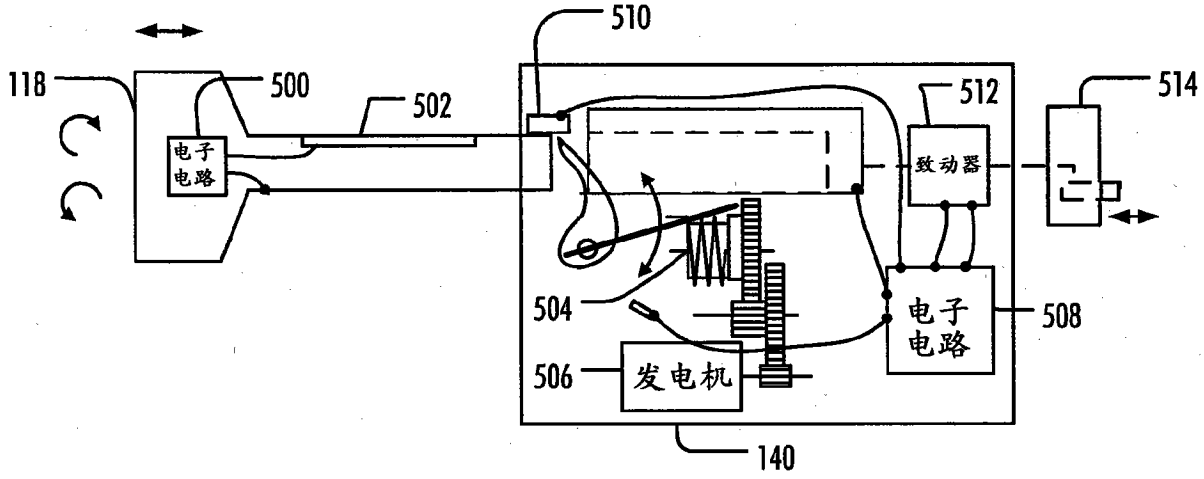

118
500
502
510
512
514
电子电路
致动器
504
506
发电机
电子电路
508
140

图 5

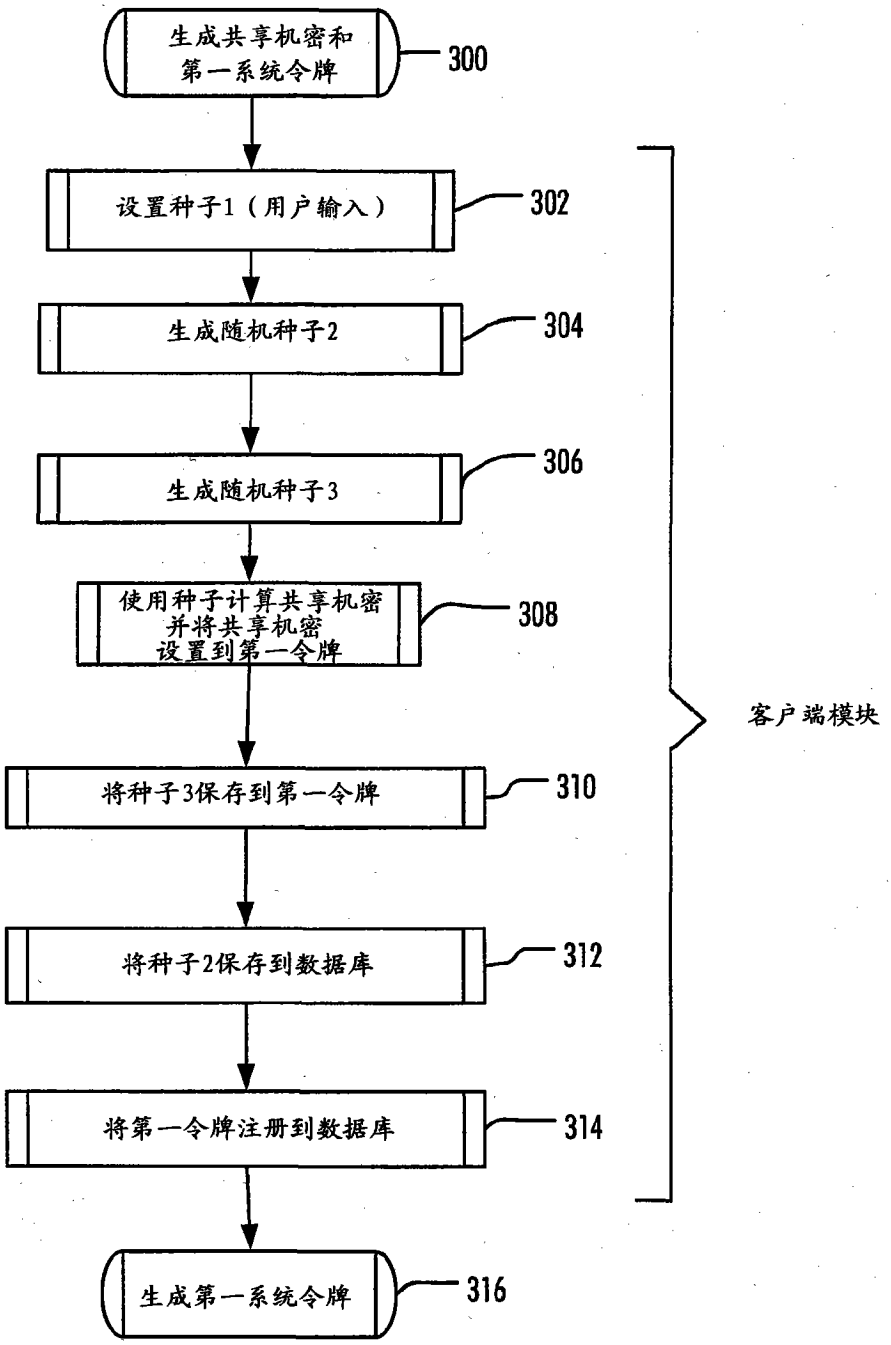

生成共享机密和第一系统令牌
300
设置种子1（用户输入）
302
生成随机种子2
304
生成随机种子3
306
使用种子计算共享机密并将共享机密设置到第一令牌
308
将种子3保存到第一令牌
310
将种子2保存到数据库
312
将第一令牌注册到数据库
314
生成第一系统令牌
316
客户端模块

图 3A

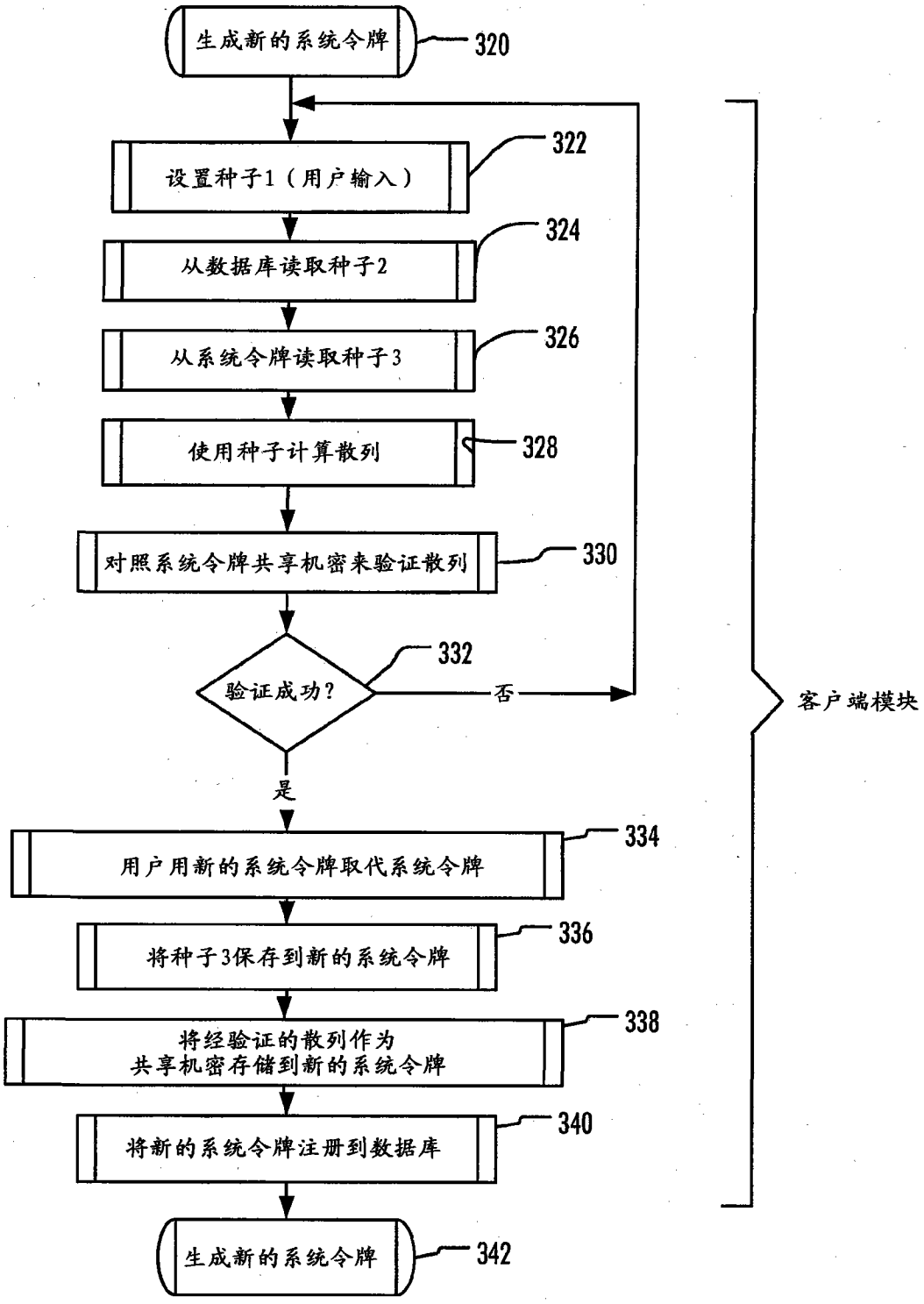

生成新的系统令牌
320
设置种子1（用户输入）
322
从数据库读取种子2
324
从系统令牌读取种子3
326
使用种子计算散列
328
对照系统令牌共享机密来验证散列
330
验证成功？
332
否
是
用户用新的系统令牌取代系统令牌
334
将种子3保存到新的系统令牌
336
将经验证的散列作为
共享机密存储到新的系统令牌
338
将新的系统令牌注册到数据库
340
生成新的系统令牌
342
客户端模块

图 3B

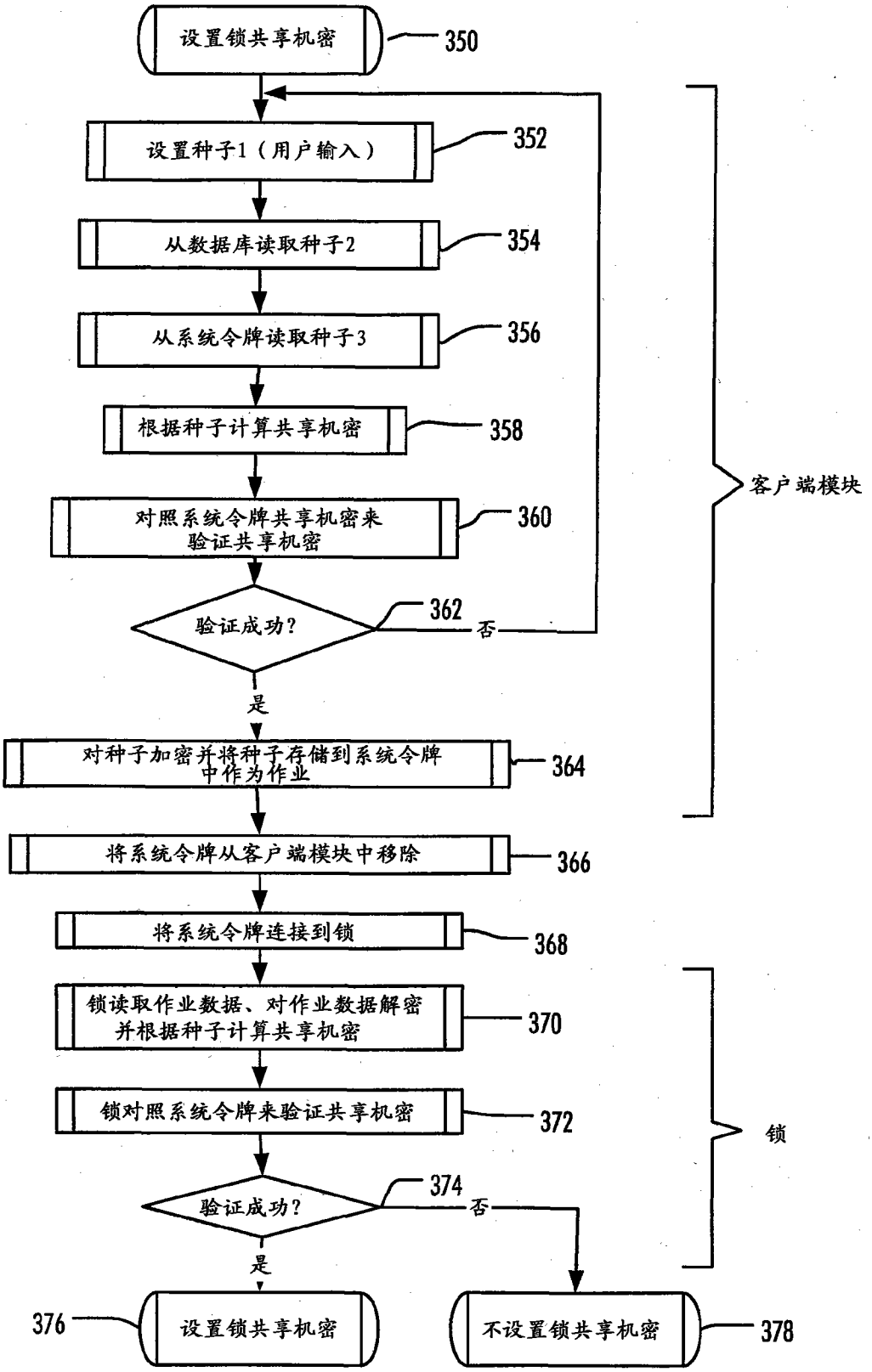

设置锁共享机密
350
设置种子1（用户输入）
352
从数据库读取种子2
354
从系统令牌读取种子3
356
根据种子计算共享机密
358
对照系统令牌共享机密来验证共享机密
360
验证成功？
362
否
是
对种子加密并将种子存储到系统令牌中作为作业
364
将系统令牌从客户端模块中移除
366
将系统令牌连接到锁
368
锁读取作业数据、对作业数据解密并根据种子计算共享机密
370
锁对照系统令牌来验证共享机密
372
验证成功？
374
否
是
设置锁共享机密
376
不设置锁共享机密
378
客户端模块
锁

图 3C

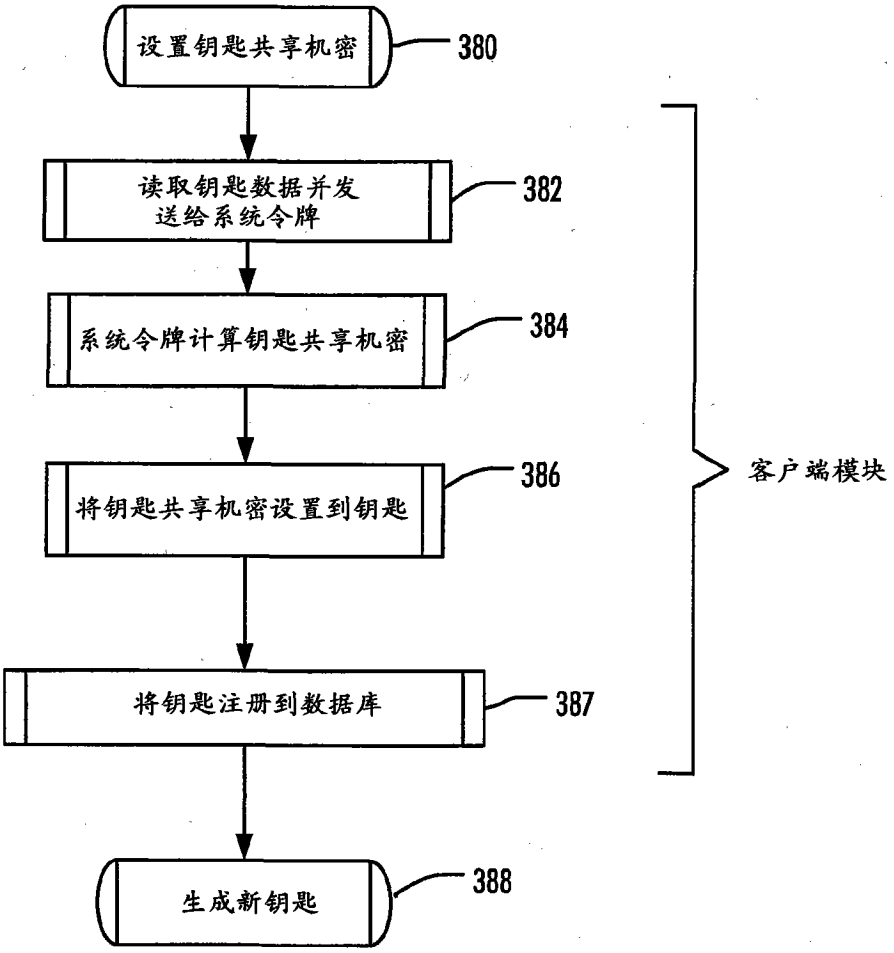

设置钥匙共享机密
380
读取钥匙数据并发送给系统令牌
382
系统令牌计算钥匙共享机密
384
将钥匙共享机密设置到钥匙
386
将钥匙注册到数据库
387
生成新钥匙
388
客户端模块

图 3D

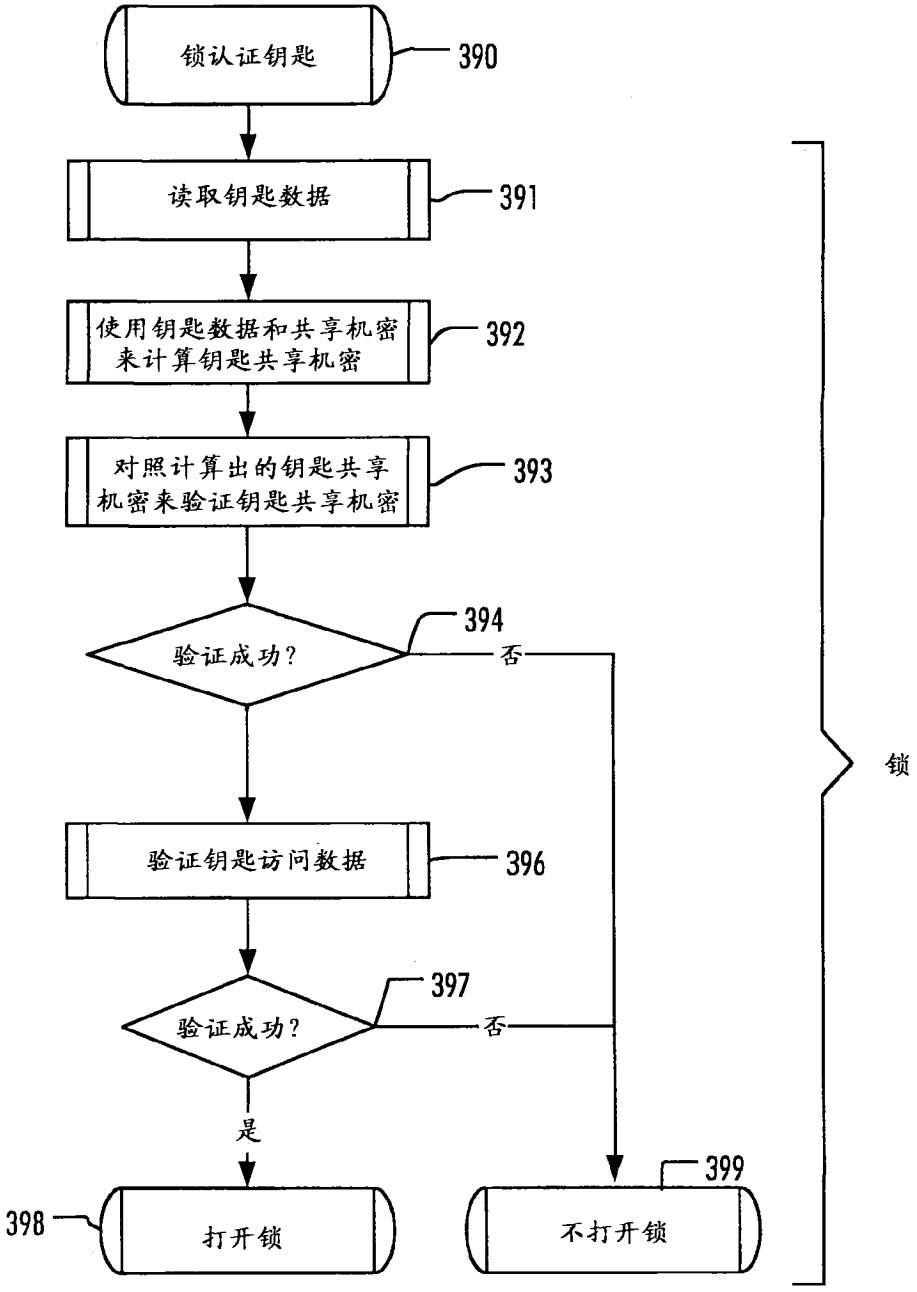

锁认证钥匙
390
读取钥匙数据
391
使用钥匙数据和共享机密来计算钥匙共享机密
392
对照计算出的钥匙共享机密来验证钥匙共享机密
393
验证成功？
394
否
验证钥匙访问数据
396
验证成功？
397
否
是
398
打开锁
399
不打开锁
锁

图 3E

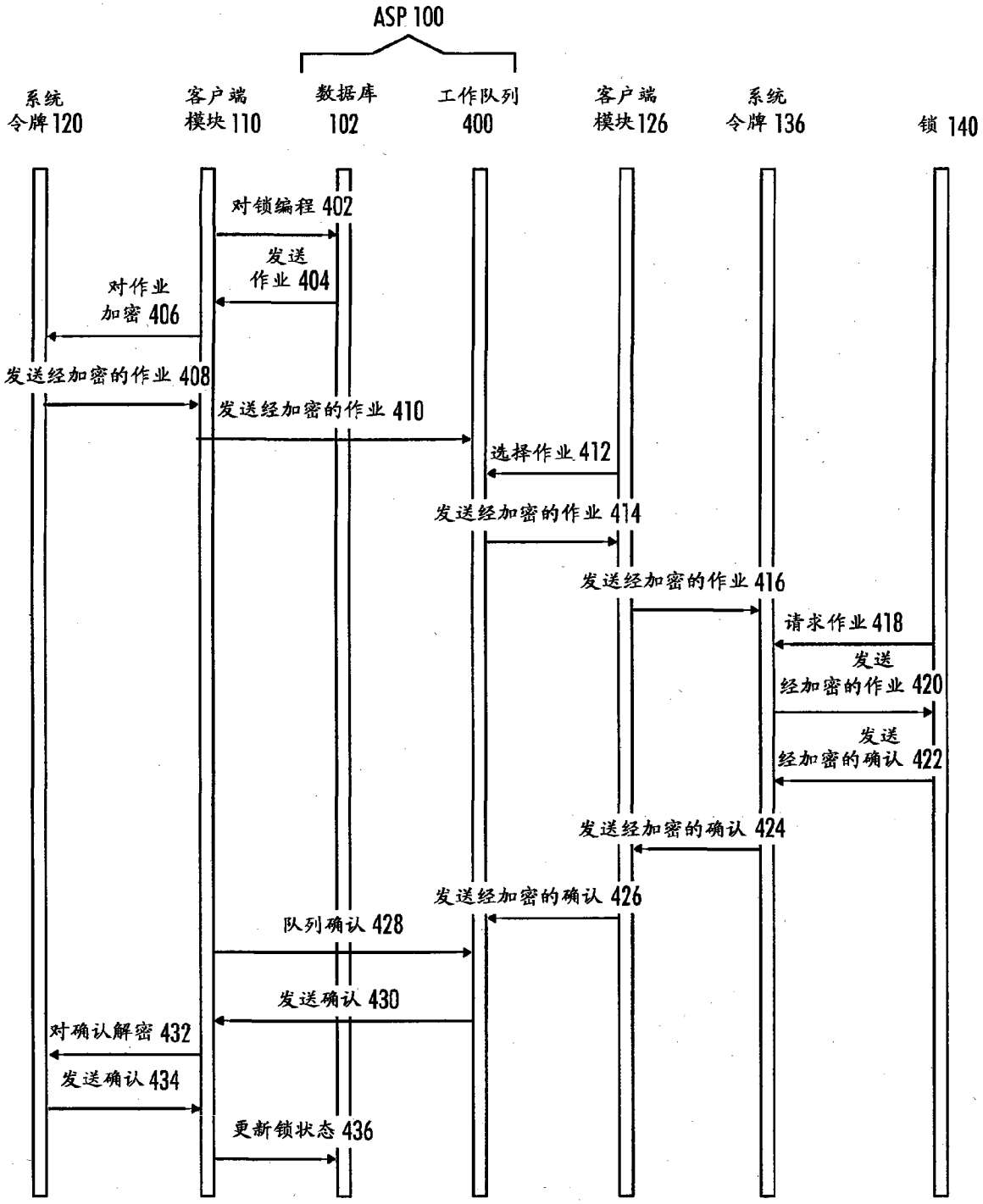

ASP 100
系统令牌 120
客户端模块 110
数据库 102
工作队列 400
客户端模块 126
系统令牌 136
锁 140
对锁编程 402
发送作业 404
对作业加密 406
发送经加密的作业 408
发送经加密的作业 410
选择作业 412
发送经加密的作业 414
发送经加密的作业 416
请求作业 418
发送经加密的作业 420
发送经加密的确认 422
发送经加密的确认 424
发送经加密的确认 426
队列确认 428
发送确认 430
对确认解密 432
发送确认 434
更新锁状态 436

图 4